AFRL-RX-WP-TR-2015-0069

OPEN-SOURCE, DISTRIBUTED COMPUTATIONAL ENVIRONMENT FOR VIRTUAL MATERIALS EXPLORATION

Marcus D. Hanwell

Kitware, Inc.

JANUARY 2015
Final Report

THIS IS A SMALL BUSINESS INNOVATION RESEARCH (SBIR) PHASE I REPORT.

AIR FORCE RESEARCH LABORATORY
MATERIALS AND MANUFACTURING DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7750
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the USAF 88th Air Base Wing (88 ABW) Public Affairs Office (PAO) and is available to the general public, including foreign nationals.

Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RX-WP-TR-2015-0069 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//Signature//

MICHAEL A. GROEBER, Project Engineer
Metals Branch
Structural Materials Division

//Signature//

DANIEL EVANS, Chief
Metals Branch
Structural Materials Division

//Signature//

ROBERT T. MARSHALL, Deputy Chief
Structural Materials Division
Materials and Manufacturing Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
January 2015	Final	10 June 2014 – 15 December 2014

4. TITLE AND SUBTITLE
OPEN-SOURCE, DISTRIBUTED COMPUTATIONAL ENVIRONMENT FOR VIRTUAL MATERIALS EXPLORATION

5a. CONTRACT NUMBER
FA8650-14-M-5044

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
65502F

6. AUTHOR(S)
Marcus Hanwell

5d. PROJECT NUMBER
3005

5e. TASK NUMBER

5f. WORK UNIT NUMBER
X0V9

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Kitware, Inc.
28 Corporate Drive
Cliften Park, NY 12065

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory
Materials and Manufacturing Directorate
Wright-Patterson Air Force Base, OH 45433-7750
Air Force Materiel Command
United States Air Force

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
AFRL/RXCM

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-RX-WP-TR-2015-0069

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Statement A. Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
This document is Distribution Statement A per contract 10 June 2014. This report contains color.
PA Case Number: 88ABW-2015-3015 ; Clearance Date: 16 Jun 2015.

14. ABSTRACT (*Maximum 200 words*)
Report developed under SBIR contract for topic F141-174-0383. The structural design community regard materials models as fixed inputs to a finite element-based design process, using materials lookup tables to provide data from experimental and/or simulations performed by others. However the accuracy of multiscale materials modeling has improved significantly over the last few decades, yet these simulations have seen little application in structural design. The project will change the paradigm by which structural design tools in finite element modeling operate by providing an interface to materials simulations, extending them to make use of simulation data, and adding support for location specific properties in single parts. These tools will enable designs that make use of materials science and engineering (processing, microstructure and performance) in the finite element model, independently triggering simulations as required, providing a smart caching strategy, to facilitate pervasive solutions that accommodate a range of structural materials. These tools will enable more efficient designs that leverage the latest advances in materials engineering, giving designers the ability to seamlessly draw on the most accurate information at every stage of design, validation and production. Integrated approaches using existing commercial tools will result structural designs making the best use of material properties.

15. SUBJECT TERMS
SBIR Phase I Report, modeling framework, multi-scale, data management

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT	b. ABSTRACT	c. THIS PAGE			Michael A. Groeber
Unclassified	Unclassified	Unclassified	SAR	30	19b. TELEPHONE NUMBER *(Include Area Code)* (937) 255-1317

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

Figure **Page**

1.0 SUMMARY

This report is the Final Technical Report for the “Open-Source, Distributed Computational Environment For Virtual Materials Exploration” addressing Phase I of SBIR topic “F141-174: Computational Tools to Virtually Explore Material's Opportunity Space from the Designer's Workstation” (under contract number FA8650-14-M-5044).

Any questions or comments pertaining to this report should be directed to:

Dr. Marcus D. Hanwell
Principal Investigator
Phone: (518) 881-4937
Fax: (518) 371-4573
Email: marcus.hanwell@kitware.com

2.0 INTRODUCTION

2.1 Background

The structural design and materials science communities typically use different approaches to model the response of materials. In general, the structural design community views material properties as constant across a given region of a solid, with simple, predetermined models that are immutable. This is in stark contrast to materials scientists who model complex materials properties, which are affected by different manufacturing processes, stress and strain wear regimes, and fatigue loading over time. They are also affected by different conditions and other factors that have serious impact on short and long term materials performance. This difference in worldview has resulted in the production of sub-optimal designs or, on occasion, even unsuccessful designs, since structural analysis codes fail to accurately model the behavior of complex components due to their use of oversimplified materials models. A vivid example of this may be found in the role of microtextured regions in titanium alloys that are not accounted for within structural design codes, yet have caused catastrophic failures of structural parts.
However, the lack of inclusion of material microstructure information in structural design codes is not a matter of negligence or choice; rather, it is because heretofore no methods existed to quantitatively and objectively acquire the necessary information and then represent and manage this information within design systems.

Fortunately, there now exist simulation codes that operate from microstructural information and are able to predict materials properties with increasing levels of accuracy, potentially enabling designers to move beyond relatively fixed design inputs, toward direct use of active material variables that can be manipulated as part of the structural design process. By adopting more advanced materials simulation, such a paradigm shift will ultimately enable the structural design process to drive materials requirements and vice versa. That achievement will enable real-time materials exploration, composite design, and/or processing options that can be adjusted alongside of conventional optimization strategies such as varying shape to accommodate load/rigidity requirements.

Once designers begin to incorporate advanced materials models, it becomes possible to tailor designs to increase longevity or to reduce the weight of a part without compromising structural integrity. For example, advanced designs could specify advanced materials processing techniques such as heat treatments in specific regions to increase local performance without the added cost of manufacturing the entire part out of an expensive material. Figure 1 shows how a designer will iterate on a design, once additional microstructural information is exposed to them in their design tools. A larger view of the fabrication of the part is exposed by looking at how the material is processed, overlaying the CAD geometry onto the processed material, and then looking at FEM analyses of the machined part with microstructural information exposed. This then extends the design process where the designer can change the material processing parameters, or the location the part is machined from, or the shape of the part in response to the microstructure of the material.

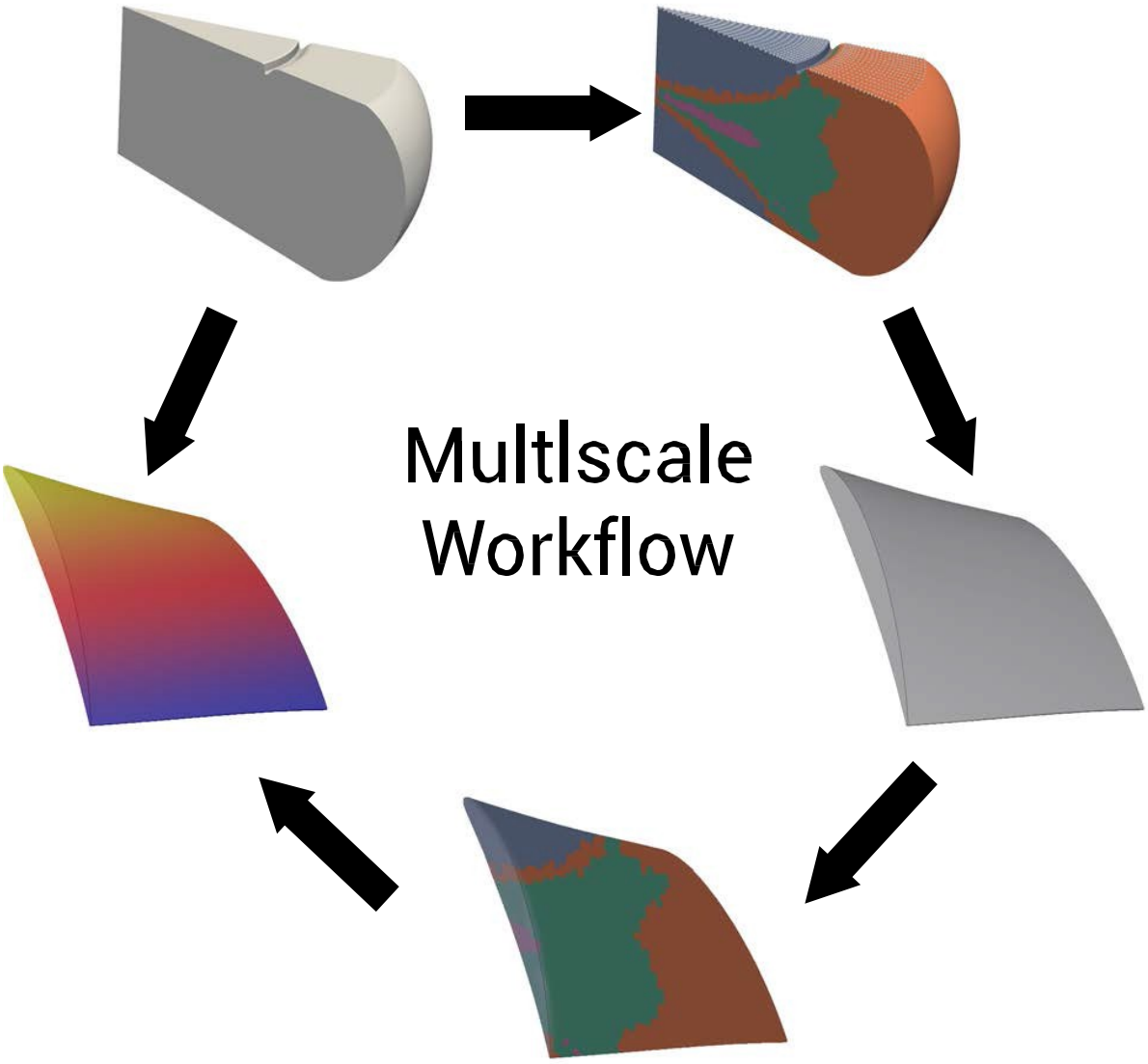

Figure 1. Overview of the Multiscale Workflow Process Explored in Phase I

In the Phase I project we researched and prototyped multiscale simulation approaches that integrates advanced materials models with a classical finite element analysis system for modeling complex systems. The goal was to determine the best approaches to achieve this integration and to demonstrate the feasibility of Phase II and follow-on commercialization. A major emphasis has been placed on identifying and designing the enabling software infrastructure that is needed to close the gaps between materials-centric simulation tools and engineering CAD-type or FEM-type design tools. Ultimately leading to the creation of a permissively licensed framework providing reference implementations for some representative design problems that can then be extended to offer an innovative environment for materials design using the best-in-class materials simulation codes to augment the process.

2.2 Approach

The challenges outlined are significant, however this also presents a great opportunity to develop new paradigms in structural design by leveraging the best-of-breed tools from finite element analysis and materials simulation codes. This requires a multidisciplinary team of experts, and the development of new work and data flows in a graphical application able to present materials parameters to structural designers. Ideally the tool should provide for varying levels of control (depending upon the domain expertise of the designer), and the sharing of both structural design and materials simulation parameters. The initial modeling, and the simulation visualization/analysis, would benefit from visualization of the material's property origins, uncertainty analysis, and integration of data from multiple sources (experimental, empirical, simulation).

This task would be too large to attempt without strong existing foundations, such as software libraries developed for both finite element analysis and materials simulation. Application development would require the integration of both components to offer a comprehensive computational environment for virtual materials exploration. The orchestration of execution of multiple standalone codes at varying length scales will need advanced high-performance computing (HPC) integration in order to offer real-time exploration and analysis using a distributed computational environment. Even running smaller, simpler simulations would require the coordination of execution of multiple independent codes, along with seamlessly moving data between codes operating at different length scales.

The coordination of multiscale modeling involves significant challenges, but also offers enormous potential to revolutionize structural design. It is clear that we are now in a time when real-time exploration is possible using distributed HPC architectures. The development of an open source, extensible computational environment that leverages codes from both domains opens up a new world of possibilities by enabling designers to manipulate additional dimensions of design previously not accessible to them. This also enables improved reuse of established materials codes by a wider range of scientists and engineers. The materials simulation codes operate at a length scale below that of the finite element models, and so there is an opportunity to provide an abstraction at that point that provides the prototype system the opportunity to demonstrate viability. Once that has been demonstrated, a platform could be deployed around this and offered to existing commercial codes along with research codes with a set of interfaces and adapters. This fits into Kitware's “Platform Strategy” where software platforms are built, and maintained by Kitware experts with a strong services model built around these open platforms. Through the use of permissive, open source licensing we will be well positioned to develop lucrative software services around the platform.

Kitware teamed with BlueQuartz Software and Sandia National Laboratories to address these challenges. The team has a history of developing and publishing open source code, and engaging with large collaborative communities to develop leading-edge computing technologies. Further the team is actively creating tools for finite element analysis, materials simulation, and materials research. The team members have identified areas of expertise that form the central pillars of the project, and combining these elements will lead to an innovative solution.

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

The Phase I focused on establishing an engineering scale baseline, and extending that to explore approaches that enable structural design to move away from seeing a material's properties as fixed inputs. The goal was to expose the material's properties as a parameter that may be adjusted to satisfy desired structural properties in a graphical design environment that fosters an integrative approach.

The main effort used simple prototypes that extended a known engineering scale baseline simulation, to explore multiscale modeling workflows. Our initial analysis, and previous experience, led to the proposal of five broad areas. The integration of multiscale simulation and design in the model builder/simulation visualization was explored, along with the possible hooks that could be used to coordinate larger workflows spanning tools developed by different groups.

The high level approach explored the following elements:

- Create an engineering scale baseline using the finite element code
- Define the interfaces between components, and the flow of data
- Modify the materials simulation code to be used in prototypes
- Develop prototypes using the finite element code with the materials simulation code:
    - Hot start method
    - Calculated table method
    - Tightly coupled method
    - Extension of multiscale to the atomic scale
    - Smart cache method
- Design an interface proposal to bridge materials code with finite element codes
- Graphical user interface design outline for the multiscale design interface
- Down select approaches, develop an implementation strategy for Phase II

These elements span different areas of expertise, length scales and software packages. The exploration of several approaches (outlined above) fed into the design of the interface specifications to guide future development. The team was composed of experts from Kitware in FEM/FEA, from the graphical model building, execution, scaling, visualization and analysis components broadly developed as part of the Computational Model Builder (CMB) and ParaView projects. The team also has experts in materials simulation, with the Avogadro, and broader Open Chemistry project, integrating with a number atomic scale simulation codes.

Kitware teamed with BlueQuartz Software in order to leverage and extend the capabilities of the DREAM.3D project in addition to their experience with the VTK and ParaView toolkits. The third team member is Sandia, a partnership that has spanned many years with Kitware, bringing significant experience in finite element modeling, specifically the Albany project. This teaming arrangement significantly augmented our ability to rapidly explore the approaches outlined above, and down select for Phase II development using some of the most advanced, open source code bases developed with research and development in mind from the outset.

3.1 Engineering Scale Baselines

The team developed two engineering scale baseline after discussions with the Air Force. The first was sketched out in the CUBIT tool, and meshed using implicit shape geometry with three distinct material zones. It approximated a jet engine turbine, with material zones reflecting microstructure of the material introduced in processing. Some preliminary development of the model looked at assigning different parameters to the shaft, and applying a displacement at the two ends. The model is shown in Figure 2, with each zone having a unique color assigned.

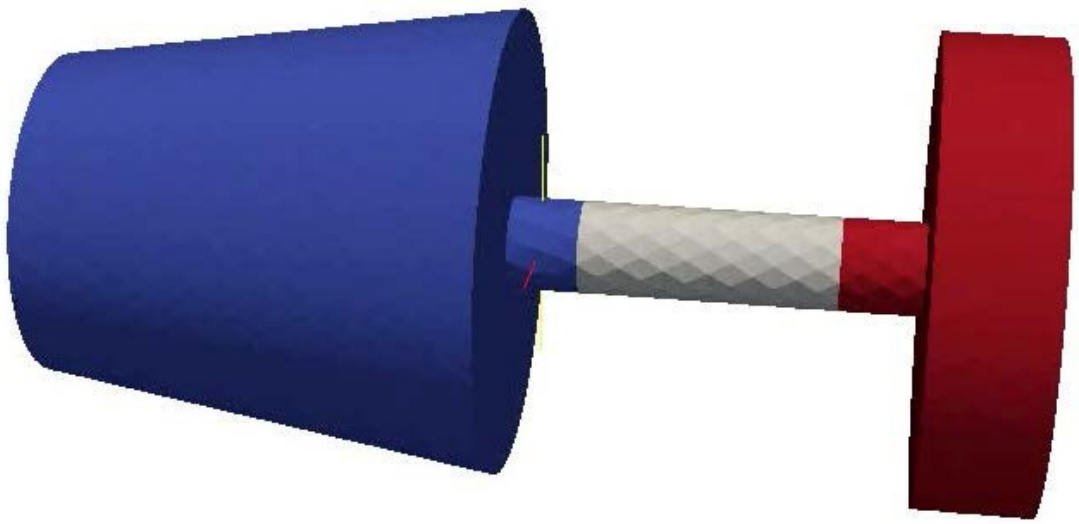

Figure 2. Engineering Scale Model - a Simple Turbine

The mesh was simplified, and was the first one developed as input to the Albany program as part of this project. Implicit surface functions were used, bearing some similarity to approaches used in CAD packages. The model was used as a starting point due to its interest in the aerospace industry, and the underlying materials processing challenges where the grain sizes, and boundary locations, are parameters that can be controlled. Some initial FEM simulations were performed using this model, and an input to Albany that defined the distinct material properties.

A second model system was also considered, and was ultimately selected as the one to carry forward. The model was derived from the output of a proprietary program called "DEFORM" that the AFRL has access to, with the input to the engineering scale baseline being the output of the DEFORM simulation. The DEFORM program effectively simulated the deformation of an alloy as it was processed using mechanical deformations. This model was selected as it more fully exposed the utility of a complex, multi-step workflow involving a number of tools that must pass data between them.

It was only possible to obtain a 2D output from the deformation simulation, but the simulation had clear radial symmetry. The supplied 2D mesh was transformed using a small, custom filter designed to take the 2D mesh and apply appropriate rotations within the CMB framework. The complete simulation process could have been performed in 2D, but it was preferable to demonstrate the capabilities of the approaches explored in 3D (Figure 3).

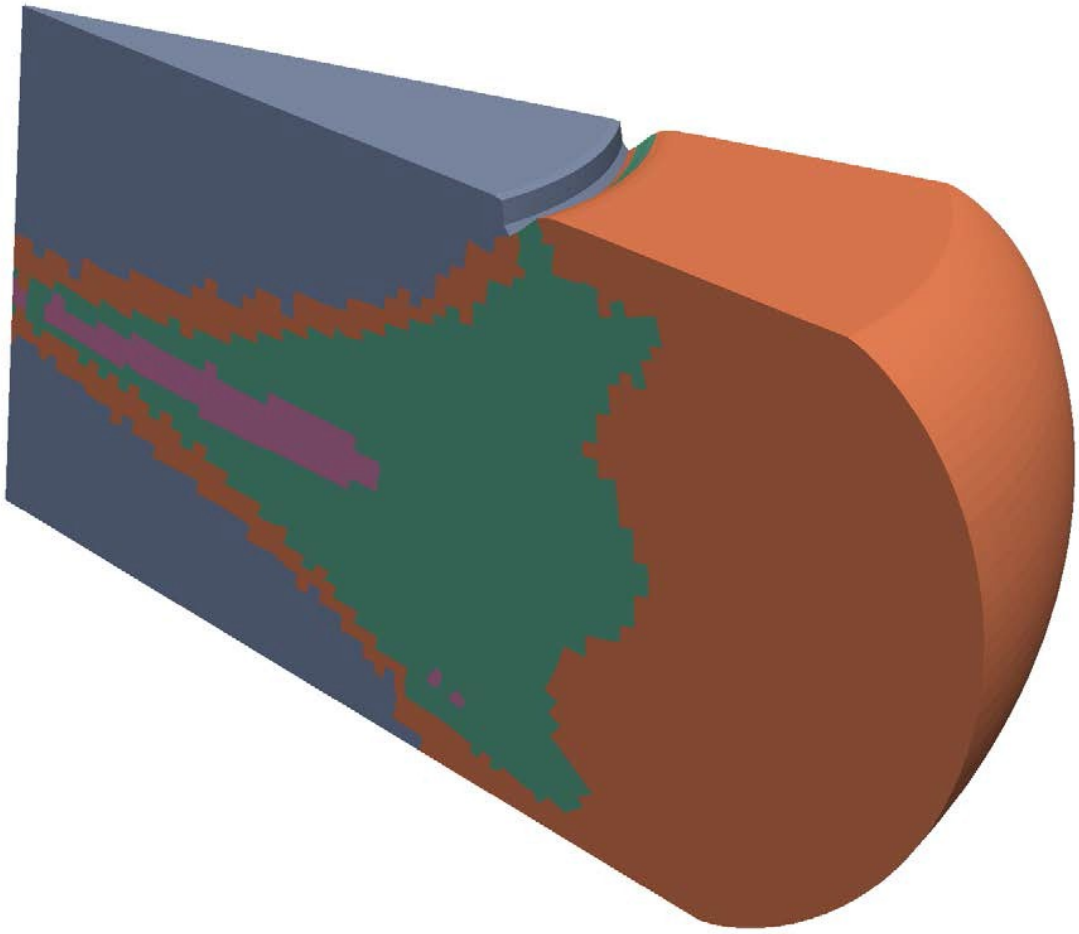

Figure 3. Zoned Wedge Extruded from the 2D DEFORM Output

The geometries were generated using two techniques, one from implicit bounding shapes meshed using CUBIT, and another using 3D extrusion to create a 3D mesh using data structures in VTK, passed through the Simulation Modeling Toolkit (SMTK) to create a mesh. Both ultimately resulted in files using the Exodus file format, which acts as the primary geometry input format for the Albany package. The methods of generation are of some importance, but the standard workflow proposed would generally defer to the CAD shapes, and the meshing capabilities present in the workflow used by the structural design engineer. The more important aspect was to generate distinct geometry that serve as input to

the prototypes, and to demonstrate the capability to transfer data between the pertinent components using the software libraries and applications.

3.2 Software Infrastructure

One of the most important aspects of the approach to multiscale modeling proposed is the transfer of data between components. A key challenge identified was not the lack of FEM codes, or the lack of materials simulation codes – it was the difficulty in using the two in an approachable, user-friendly software environment. It became clear as the approaches were explored that integrated, user-friendly, and extensible software infrastructure would provide a valuable software platform, that could serve as a strong basis for software services.

The data flow shown in Figure **4** shows a high level view of the steps involved in a very simple multiscale workflow, where it is clear that it is essential that a number of distinct programs must be used. At each of those steps there are translations between data formats, individual pipelines within components to perform the high level task required to provide input to the next step. The FEM solver and materials codes will often be run iteratively until some convergence or acceptance criterion is hit, where this is often dictated by the goals of the multiscale modeling task undertaken.

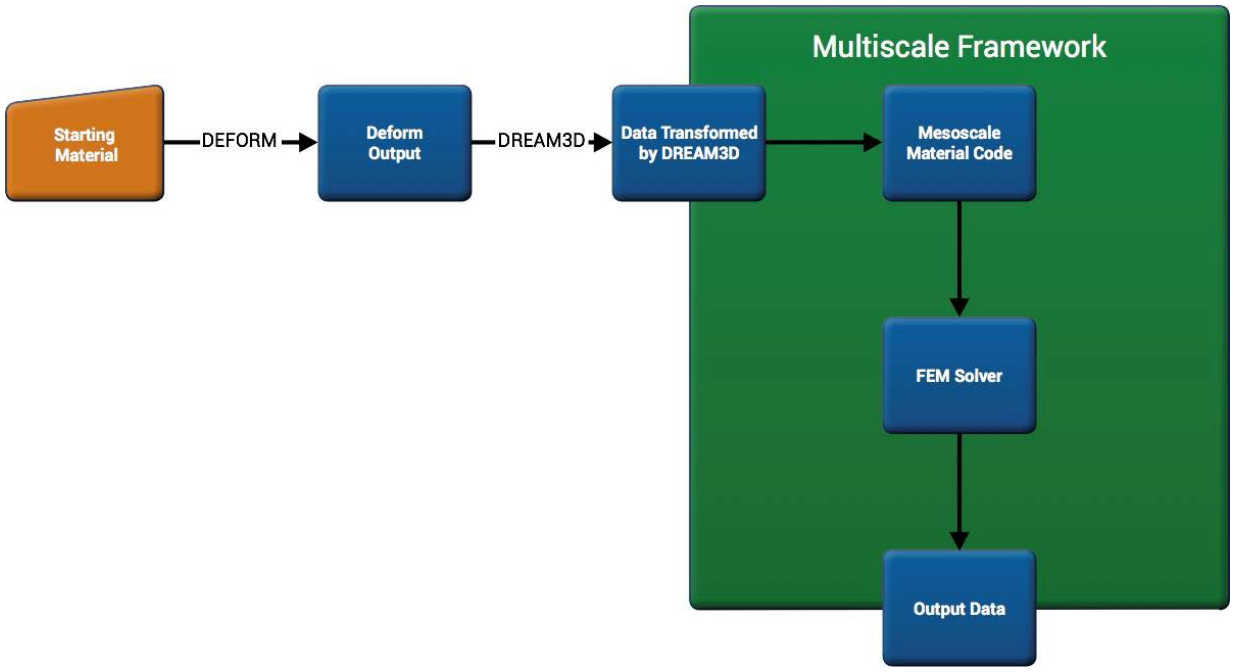

Figure 4. The Software Framework for Multiscale Modeling

The development of suitable abstractions between components in the system not only provides a platform for demonstrating capabilities, but also provides a platform for accelerated innovation. The

design and development of such a framework would offer significant opportunities for commercial activity, and enhance the value of materials codes heretofore unused in structural materials design. If microstructure from simulation is integrated, it also becomes possible to take experimentally obtained microstructure and use that in the same simulations. The layer must translate file formats, as well as parameters from the different length scales, such as temperatures from the FEM analysis that can be applied to the mesoscale. The mesoscale simulation can be used to simulate properties such as grain evolution, and the results taken and translated to FEM scale parameters such as material elasticity.

The framework for multiscale simulation shown in Figure 4 needs an interface to handle the communication between the FEM simulation and the materials simulations. The FEM simulation normally uses fixed material properties, and the goal is to move beyond that to use materials simulations. In addition to materials simulations as input it is desirable to use experimentally determined properties, and empirically determined properties, as part of simulations. The materials simulations may also take a large amount of time, leading to a need to cache simulated material properties values that will often be reused in multiple steps. Figure 5 shows at a high level proposed components in a “smart materials cache”.

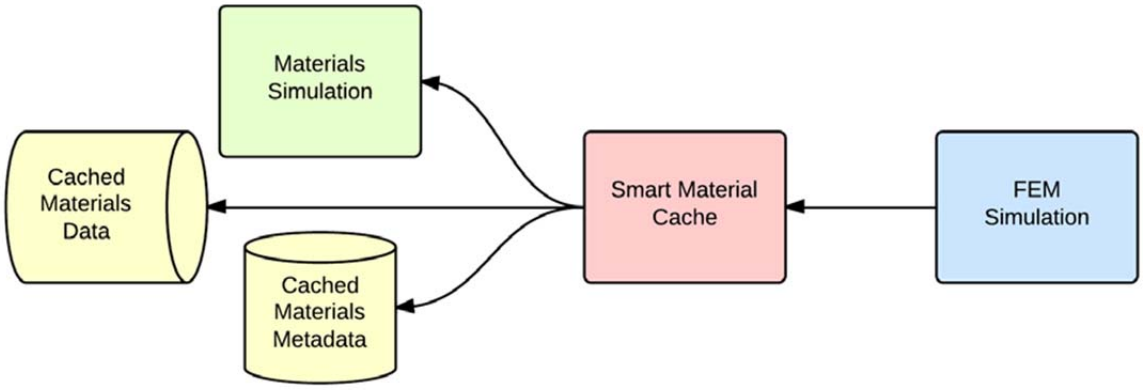

Figure 5. Design of a Smart Materials Cache for Multiscale Modeling

This element forms the core of the software infrastructure, and the approaches that would be implemented in such a cache were explored in Phase I. The cache becomes the main intermediary, and either drives the FEM simulation, or is driven by it taking queries from the FEM’s main simulation loop. It also generates the inputs to the materials simulation code, possibly initiating many parallel jobs on a HPC resource, or serially executing them. If the input matches a previous result, the material simulation can be skipped and cached values used. This approach can also be used to inject known values obtained in other ways, such as experiment or more exhaustive simulations run manually.

This design also feeds into the desire to develop an enhanced graphical environment for structural design, moving beyond the view of fixed, uniform material properties. Even when considering commercial off the shelf software it is possible to extend the design interface with additional tools for

specialized workflows. This leads to the desire to provide abstractions in the multiscale framework for interaction with graphical tools, enabling designers to go from a high level classical view of a design, to the multiple material zones used to represent microscale structure. There is also a desire to go right down to the microscale material simulation, especially in the case of structural engineers iterating on designs, working with materials scientists to improve material simulations. This may involve moving beyond generated inputs to manually modified inputs after consultation with experts on particular processing techniques.

3.3 Graphical Multiscale Design Interface

One of the major challenges in multiscale modeling, and a reason it has found limited use thus far, is the lack of support for it in commercial off the shelf software products. There is a need to provide a software framework to perform the simulations, and once that is developed there is also a need to offer graphical interfaces that can be exposed in existing design tools. This is where tools such as the Computational Model Builder (CMB) can be augmented to provide a graphical interface for multiscale design, and either extended for analysis of the outputs or interfaced with other relevant tools such as existing commercial tools and/or ParaView.

Figure 6 shows a high level overview of tools developed and/or integrated by Kitware that already provide a graphical workflow. This can be used in an independent workflow, or components of it can be integrated with other tools in order to augment tools already familiar to engineers. The graphical tools developed by Kitware are done so with reuse in mind, and integrate a number of tools developed by other groups already. The ReMUs project provides integration with a number of meshers, providing an abstraction to the graphical environment and potentially running the meshing tasks remotely. The multiscale framework can clearly be added into such a suite of tools, and exposed to offer multiscale modeling in this environment.

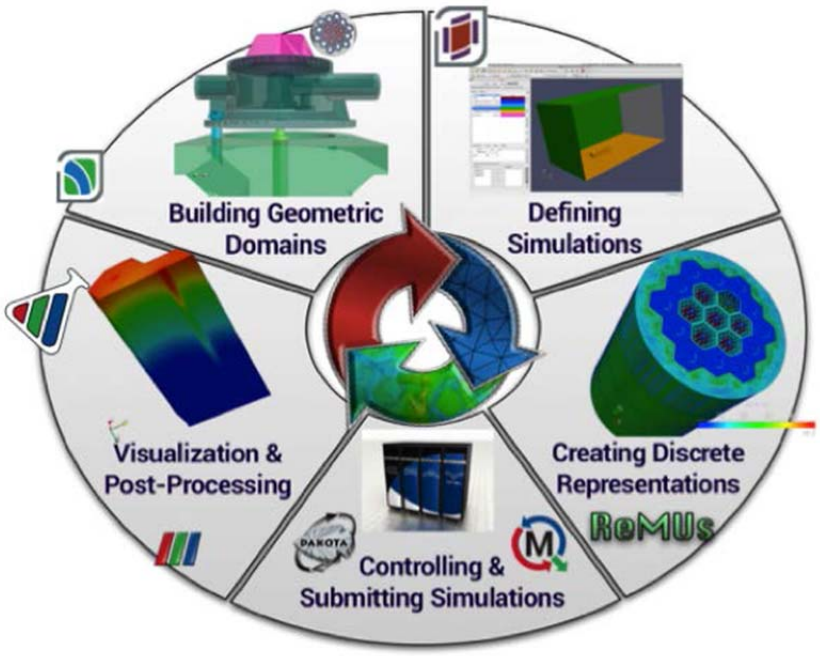

Figure 6. The FEM Scale Workflows Offered by Kitware Tools

The CMB tool provides some integration with Albany, and other codes such as Hydra. It features an extensible framework to add new simulation engines, which can be used to extend multiscale workflows, and higher-level views of pipelines generated for tools such as DREAM.3D (which itself contains an interface for building pipelines, but was also extended to support batch execution of pipelines).

The DREAM.3D project provides a number of materials data processing pipelines, with the tool being used by the materials community. It offers a simple, linear data processing pipeline, with the possibility to import and export the pipelines. The export format uses a simple text format, and so it is amenable to being manipulated in a higher-level tool. The addition of a batch oriented data pipeline processing interface makes it simpler to add DREAM.3D to larger workflows, with the option of reusing the C++ software library if tighter integration were desired in the future.

Going to the atomic scale, the Avogadro 2 project, developed by Kitware, offers generation of input for a number of simulation codes, with an extensible set of generators employing a similar mechanism to CMB. It is developed as a set of software libraries, and an application with a defined set of APIs that can be called over local sockets. The application has been demonstrated to scale to systems with millions of atoms/particles, offering facilities to push up toward large systems that can serve as input to the FEM scale simulation. Example visualization shown in Figure 7 show typical visualizations available in the library/application.

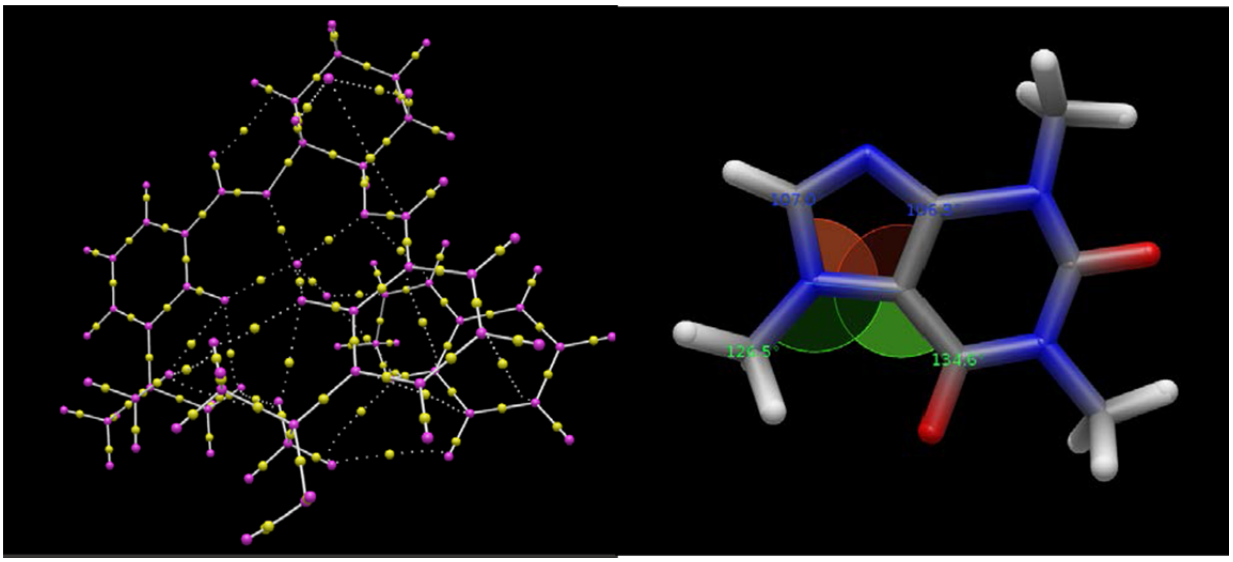

Figure 7. Atomic Visualization and Analysis in Avogadro 2

The Phase I focused on manually running multiple steps in the distinct packages, examining available integration points that will be used as part of the Phase II project. The graphical tools described all use permissive open source licenses, and can be integrated to offer a standalone environment or exposed via extensions to existing design environments. The significant development efforts already made in the tools described can be leveraged in a multiscale environment.

The tools described were all developed using the C++ programming language, using Qt and OpenGL for visual elements, offering dynamic runtime extension using plugins, and offering reusable software libraries that expose features already well tested in the main application interfaces. The challenge in the project is moving data between components and length scales, where providing reference implementations offer demonstration of capabilities and examples of how other tools might be integrated.

4.0 RESULTS AND DISCUSSION

This section presents the results of our preliminary work in Phase I, and then discusses the impact on our design for the Phase II system. It discusses some of the key challenges encountered, and how they could be addressed.

4.1 Use Cases to Develop Representative Models

The field of multiscale material modeling is vast, and it is important to narrow down the focus of the Phase I approach to a representative model that is important to key customers. This serves as a concrete model that can be used in prototyping approaches, with simplifications made to reduce the computational power required. The models were developed in 3D in collaboration with AFRL scientists to offer a compelling use case. Two possible use cases were discussed at the initial kickoff meeting, with both explored initially.

The diagram in Figure **8** shows a high level overview of the material processing use case put forward as one possibility. This system looks at the evolution of the material grain microstructure as a deformation is applied using the "DEFORM" program. This phase of the project treated the deformation step as a "black box" with no direct access to the proprietary program used to simulate deformation. It would be possible to drive this step if access were available, but the process should be flexible enough to perform steps externally too.

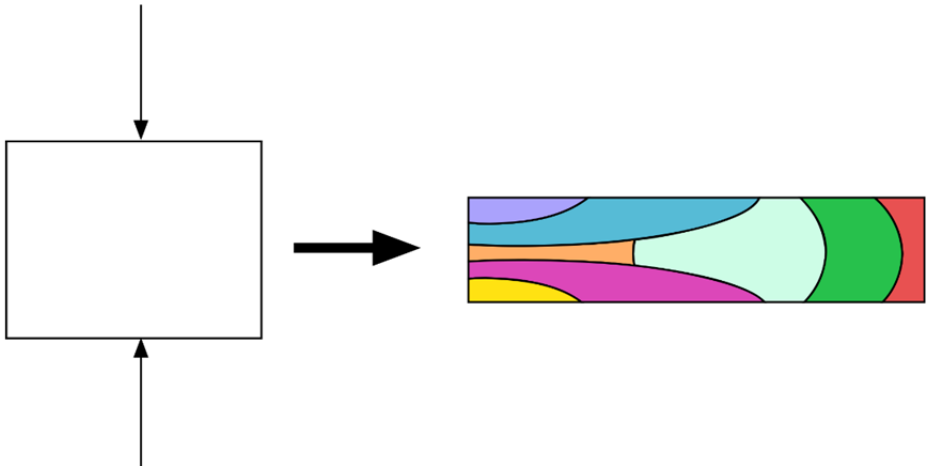

Figure 8. Deformation and Zoned Material Processing

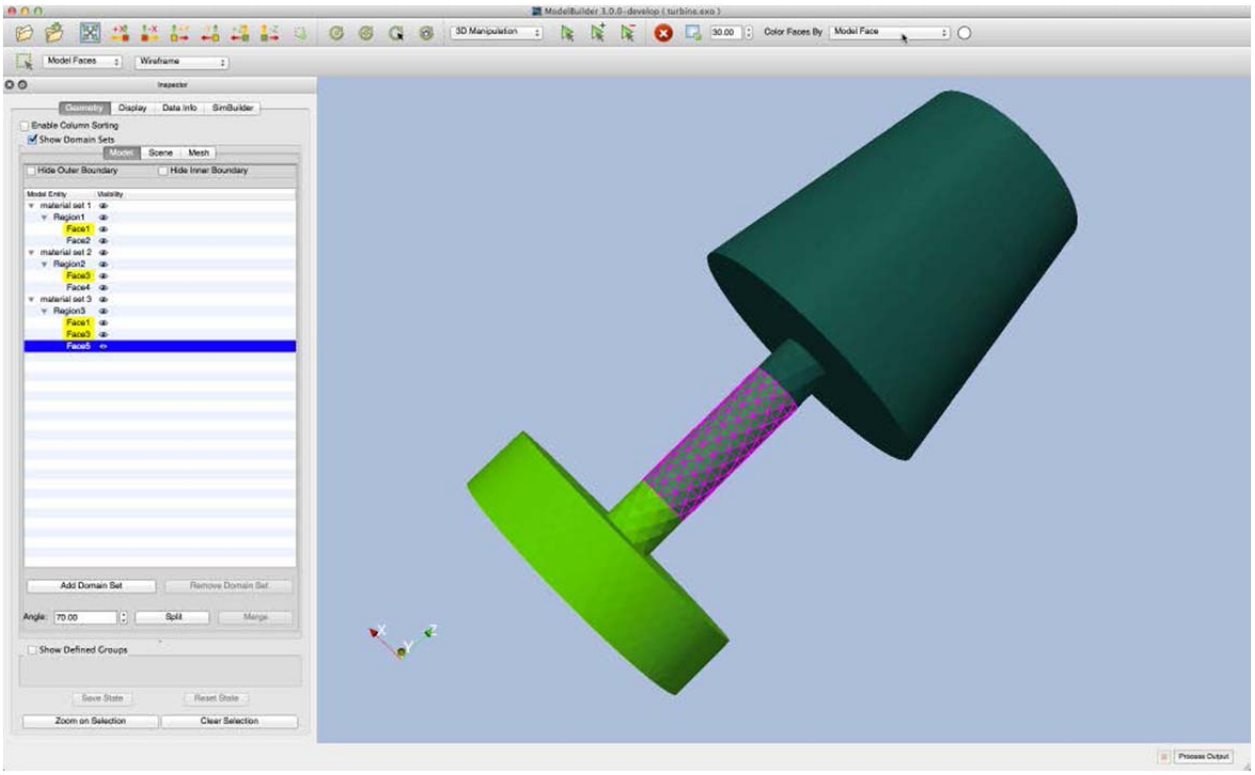

Figure 9. The Computational Model Builder Showing Three Zone Model

A simple turbine model with three distinct zones was created using the CUBIT tool, and saved as an Exodus mesh. The Exodus format is the primary input format for the Albany FEM code, and is supported by the CMB application (and many other codes). Figure 9 shows the model loaded into CMB, with the zones and model parameters displayed in the left-hand panel. Figure 10 shows the same model, but with the solid mesh hidden and interface boundaries shown. This clearly demonstrates the capabilities in the CMB application for visualizing and annotating meshes with multiple zones.

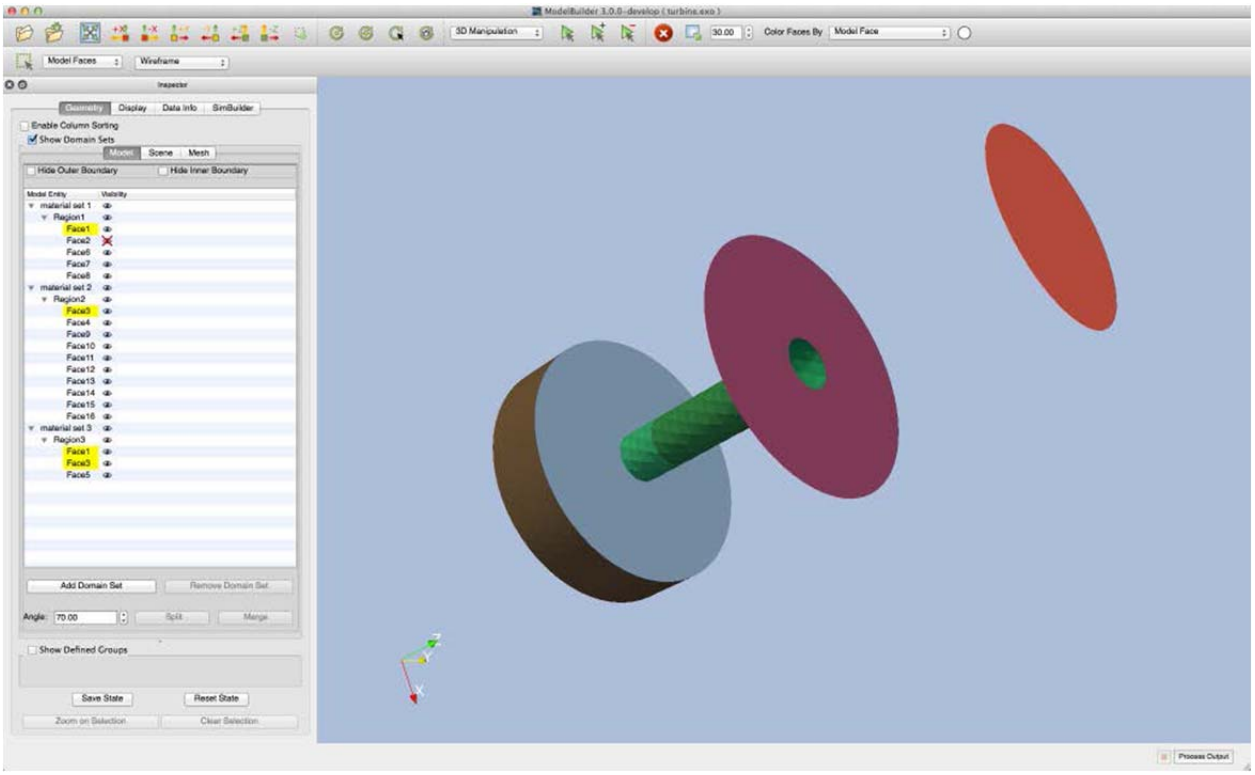

Figure 10. The Three Zone Turbine Model Showing Interface Boundaries

Each of the regions is maintained throughout the FEM simulation, and can have distinct properties defined. This simple three-region model was used in an Albany simulation, where the material properties were varied and the model was deformed. The interface at one end was fixed, and at the other end the interface was moved at each time step. The calculation input file was created, and material properties defined in the file. This file has all of the components necessary for the hot start method described in the proposal, and discussion of modifications to Albany needed to implement the calculated table method gave a clear path forward. The resulting simulation output was viewed in ParaView, and deformation could be visualized at each time step, along with calculation output parameters.

It became clear that there were some minor issues with names zones in the Exodus file not showing up as expected in the interface, Figure 11 shows the named zones in the panel on the right of the application. It is important that metadata such as this be maintained in the data pipelines, and displayed in the graphical environment. These parameters are often used in input files, and provide more meaningful references to the user of the design tool. Interaction with CAD geometry is important; Figure 12 shows a standard CAD geometry loaded into CMB, showing the named entities that come from the CAD file.

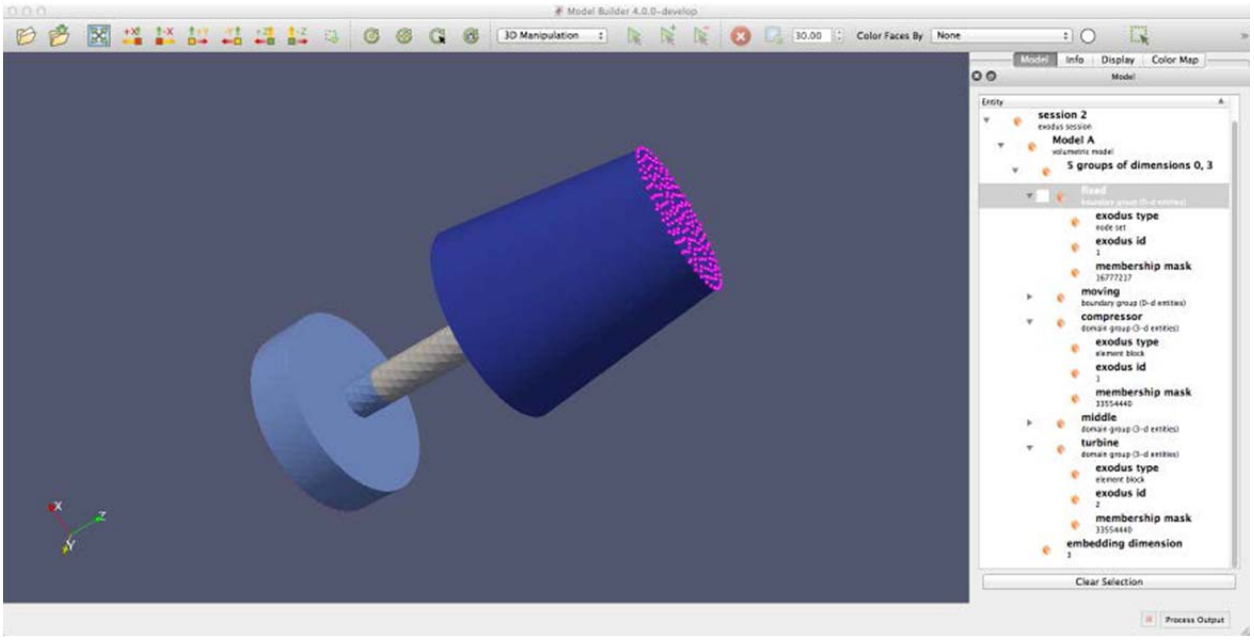

Figure 11. Named Zones Displayed in the Computational Model Builder

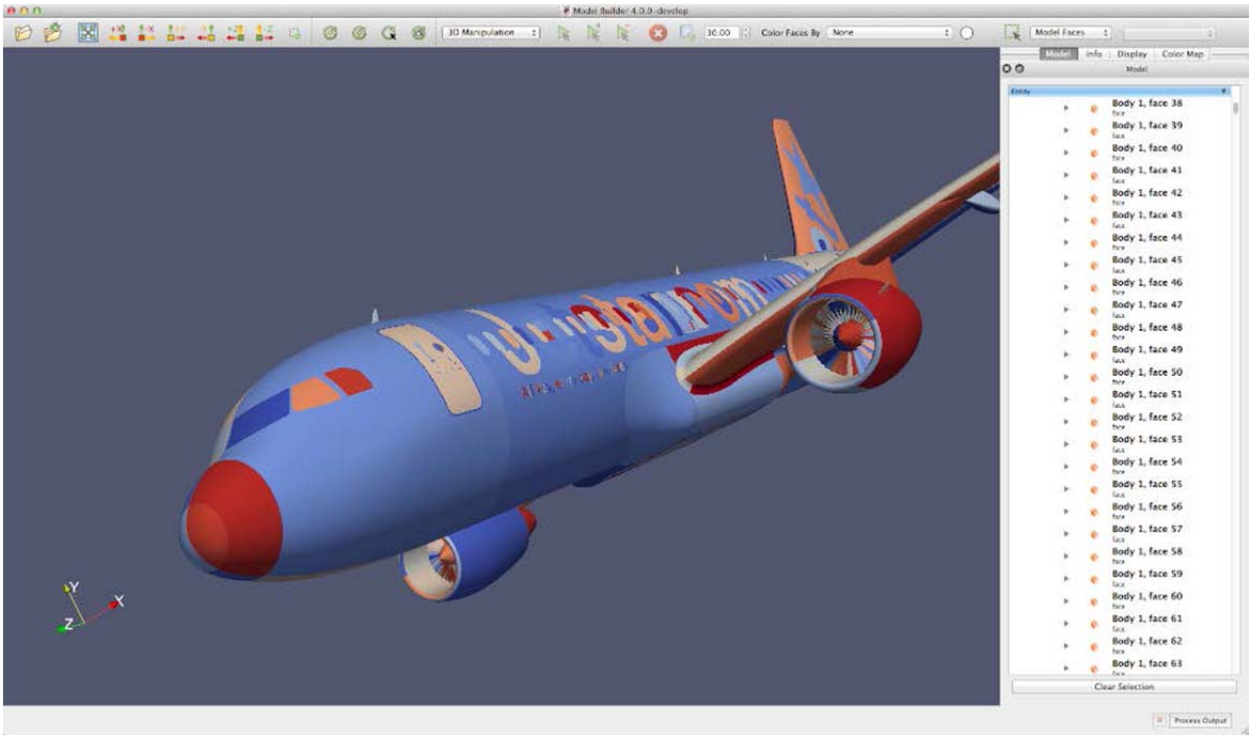

Figure 12. CAD Geometry Displayed in the Computational Model Builder

This represents one way of interacting with CAD geometry, offering a strong starting point for integration with larger existing CAD design platforms. The other source of geometry that must be considered is input from other material processing simulation tools, such as DEFORM in material deformation processing techniques. For a full solution both are necessary, the CAD geometry must be overlaid, and correlated with a processed material mesh in order to assign microstructure and other properties that may be available.

The output of the DEFORM step was a 2D mesh, and a point tracking file along with various materials parameters. Figure **13** shows the extruded 3D mesh as it appears with no processing, the point tracking file and material parameters must be processed in order to gain more useful information. A DREAM.3D pipeline, and (currently proprietary) C++ plugin was used to take the point-tracking file and perform clustering operations on the data. The pipeline was exported as a text file, simple keyword replacement was applied to take an input file, and save an output file that could be loaded by CMB/ParaView. The number of zones was another free parameter that was identified, this could be modified and the full pipeline was run with four and six zones in our experimental workflows.

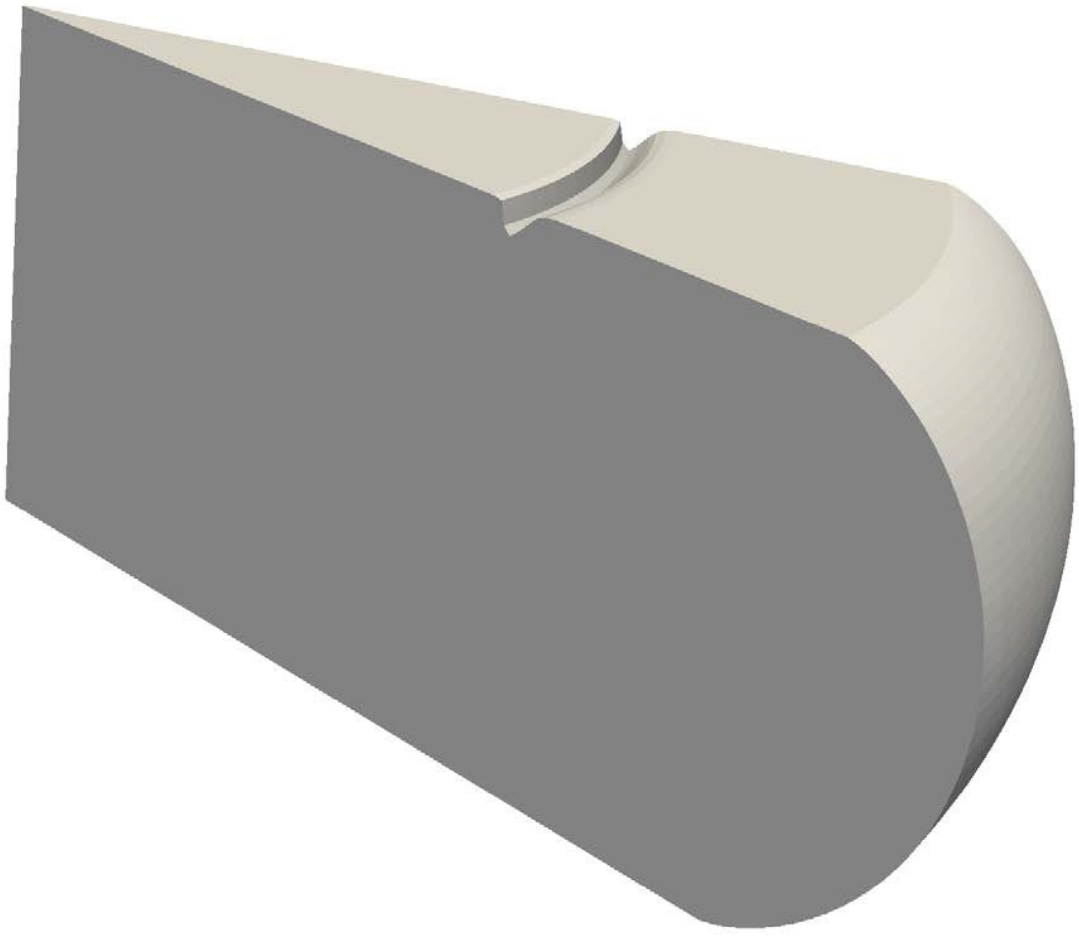

Figure 13. Wedge Geometry of Processed Material

The zoned geometry is shown in Figure 14, where the version on the right also shows a point set. This geometry was generated by taking the processed output from the DREAM.3D pipeline, rotating the 2D mesh about the axis of symmetry, using a step size and sweep angle to give a wedge. The full “pancake” would be formed by taking a 360° sweep, where smaller sweeps offer a fuller view of the internal structure.

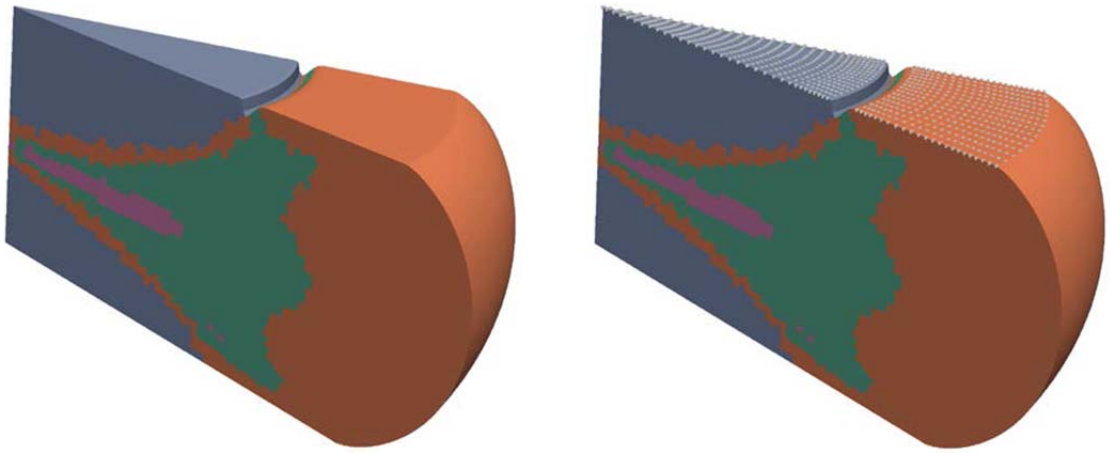

Figure 14. Wedge Geometry Processed with DREAM.3D Showing Zones

The geometry was generated using VTK data structures in some custom processing code developed for the Phase I project. It was necessary to make use of the Simulation Modeling Toolkit (SMTK) to take the mesh, and use integration with the Exodus writer to create output in a format suitable for Albany. The point set shown on the top of the geometry in Figure 14 was also exported in an additional operation using SMTK functionality, as were 2D boundaries on the different faces. These could all be exposed in custom graphical elements, but for the purposes of the Phase I project prototypes were developed and executed using simple executable programs that could be called from the command line.

Once the full geometry has been generated it can be loaded into CMB or ParaView. Figure 15 shows the geometry loaded into the CMB application, with the different named zones, and some zones hidden. This processed, zoned mesh is the input for location specific heat treatment use cases shown in Figure 4, where this mesh has a number of zones representing a set of microstructures that were obtained either experimentally or synthetically. For the purposes of the Phase I development a DREAM3D pipeline was developed to generate a synthetic microstructure. This development included updates to several modules including the input statistics generation; SPPARKS file writing and general bug fixes. The final output of the DREAM3D pipeline was a SPPARKS file that can be used as inputs into the next step of the process.

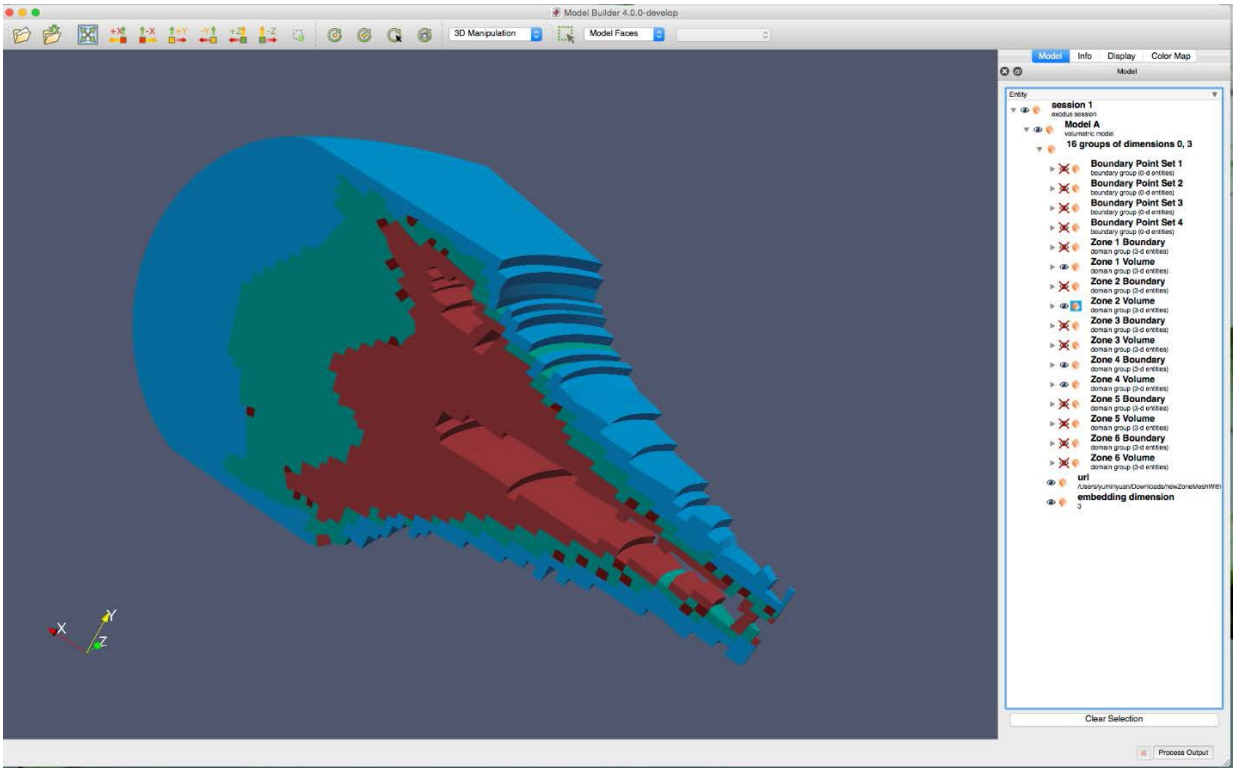

Figure 15. The Zoned Wedge Geometry in Computational Model Builder

The final step in prototyping the model system was to run the Albany-SPPARKS multiscale simulation, taking the input model parameters generated in the previous steps as inputs. This step proved more challenging than initially thought, and ultimately we refocused some of our efforts on other components and did not get a working prototype of the tightly coupled method working. The interfaces within the Albany code were explored, with an existing coupling providing a very basic initial implementation. It highlighted the need to remain flexible on the level of integration between the codes, although with suitable abstractions in a multiscale framework it is clear that hot start methods can be developed quite rapidly, and deeper integrations implemented depending upon requirements.

A deeper understanding of the Albany simulation group was developed during the course of the Phase I project, and the level of interaction required. It is clear that the implementation of a smart cache would provide an ideal interface that could be queried from Albany's simulation loop for updated material parameters. This would remove the need to implement things like parameter caching from the FEM code, placing them in an independent and reusable framework that could also be leveraged from commercial FEM solvers. The hot start method was coordinated using Python scripts and input files, and would serve as an initial prototype to be implemented in the framework proposed.

The SPPARKS code, also developed at Sandia, provided a mesoscale materials simulation code that could be used in the location specific heat treatment prototype. It had existing notions of grain orientation, evolution, and so on, along with macroscale parameters that can be passed to/from the FEM solver. The individual simulations were relatively fast, offering a simple platform for prototyping. It takes a text

input file typical of codes developed in this area, with some existing support in DREAM.3D that was extended for this model. Figure 16 shows the analysis of an early Albany simulation (left), with the zoned input (right).

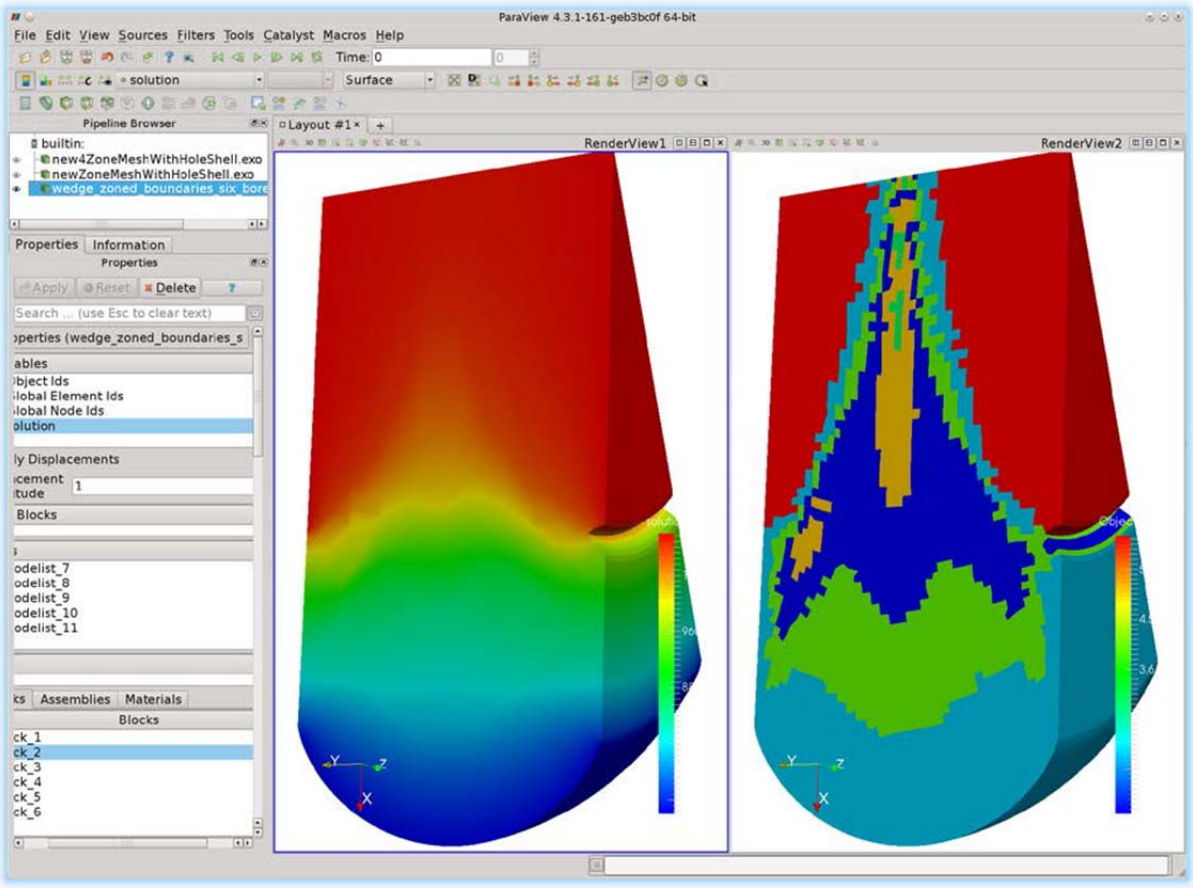

Figure 16. ParaView Showing Analysis of Albany Simulation

4.2 Additional Use Cases

During the Phase I project the team worked to identify additional use cases, along with how far the current use case should be taken in follow up development. The use case dealing with location specific heat treatment currently ends with the multiscale simulation of location specific heat treatment. This can be taken further, adding integration with CAD design tools to assign microstructure to the CAD mesh, and optimize placement in the processed material. In addition the placement of blanks in the material, used to verify material-processing steps succeeded as expected. Figure 1 shows the high level iterative design cycle proposed, where the analysis of the machined part would also benefit from integration of multiscale modeling.

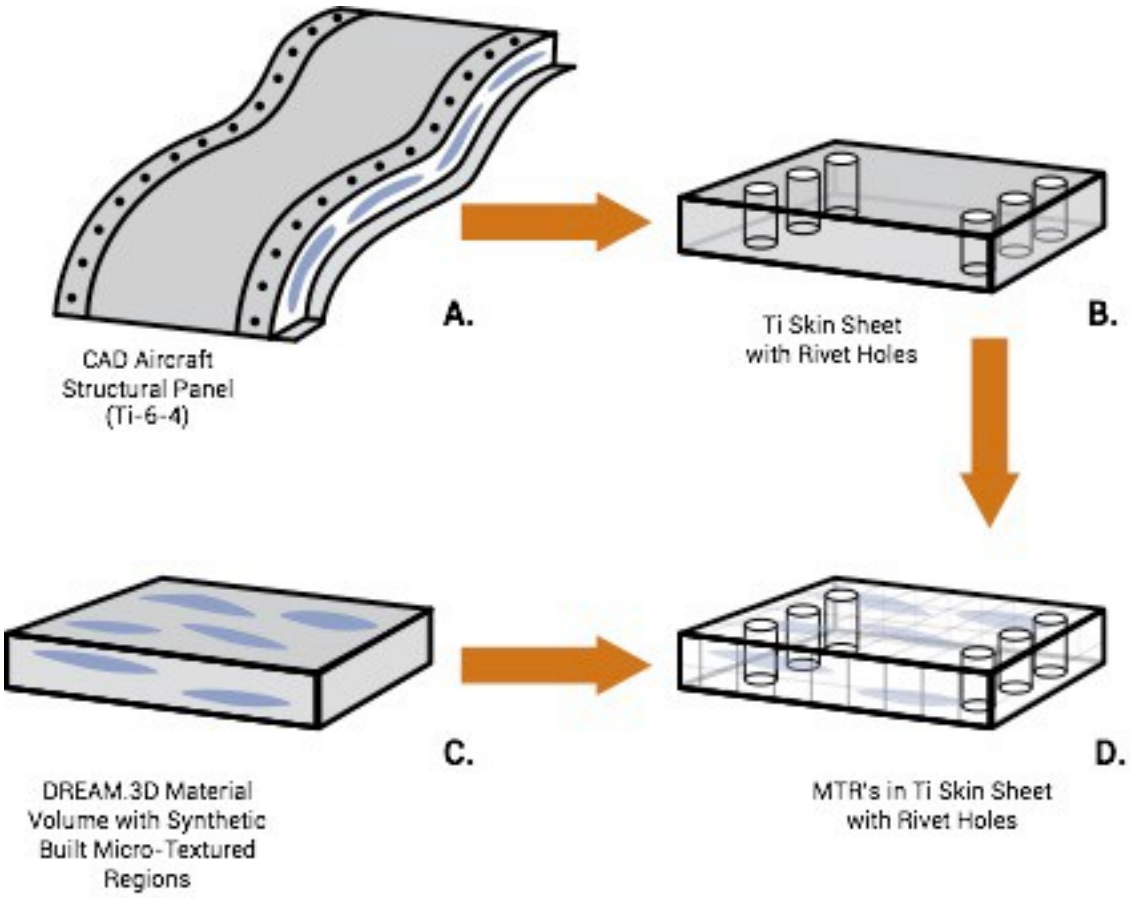

Figure 17. Proposed Micro-Textured Region Use Case

Another use case that bears a number of similarities to the one described above is shown in **Figure 17**, working with micro-textured regions that are used in aircraft structural panels. The CAD intersection, placement of rivet holes, and location of micro-textured regions offer an important use case that could reuse many of the same components developed for the first use case.

A final use case outlined in Figure 18 links right down to the atomic resolution, with grain volumes as intermediate (bearing similarities to the previous two use cases). This opens up the possibility of applying more accurate methods where appropriate, and aiding in the design of nano- or micro-objects. Codes such as LAMMPS already have some integration layers that take the discrete simulation and produce continuum parameters that can be used by FEM solvers. This would act as a great stretch use case where the generality of the multiscale framework could be improved, and demonstrated if sufficient development time exists. The major challenges being the simulation times involved for codes like LAMMPS with big enough systems, and the link between molecular dynamics materials simulations and FEM parameters.

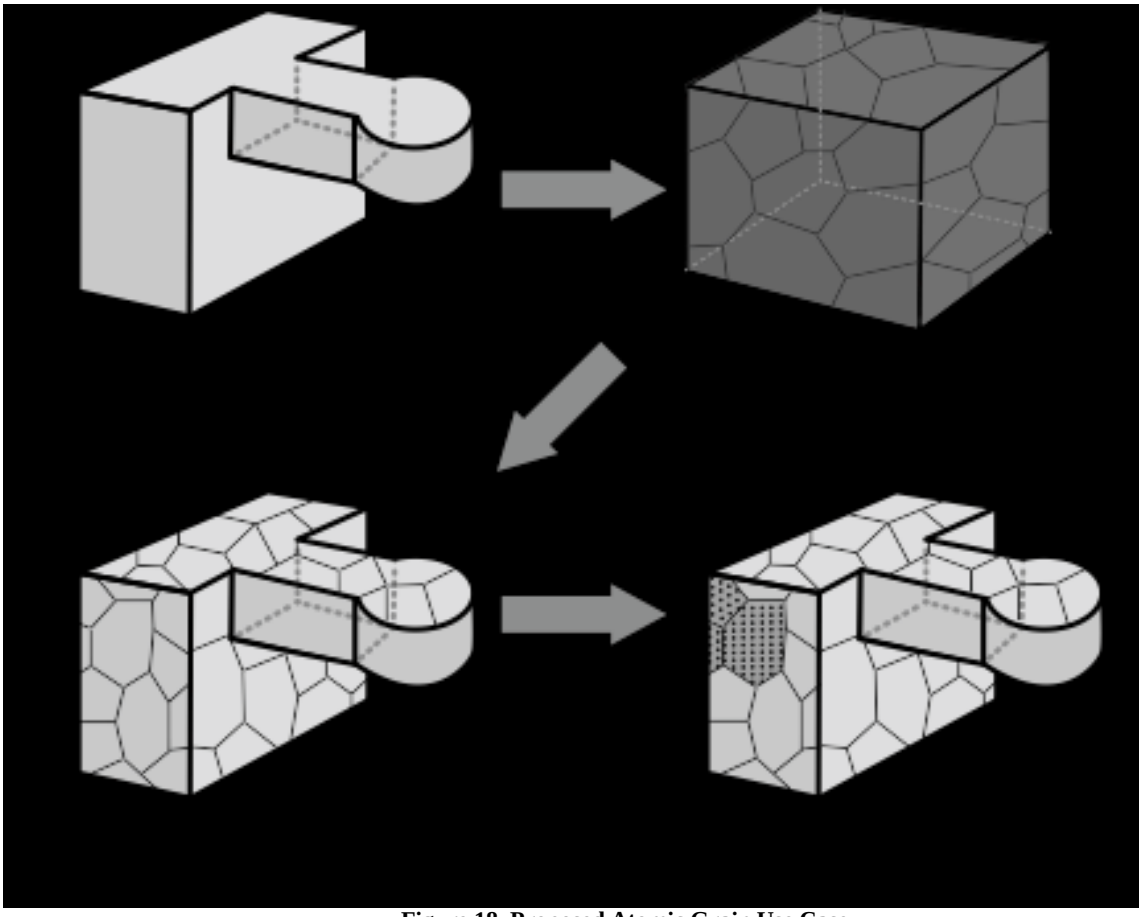

Figure 18. Proposed Atomic Grain Use Case

4.3 Partnership with Commercial FEM Codes

It was clear from the call, and after discussions with AFRL staff, that integration with existing commercial FEM codes was desirable. After discussing the dominant platforms, and performing an analysis of the available partnership programs Kitware approached the partnership managers for the ABAQUS and ANSYS codes. Both codes offer partnership programs, with generous licensing for companies developing integrated solutions with their products.

It is clear that these packages, despite expensive licensing fees, offer the best platform to develop enhanced multiscale modeling solutions with. After consideration of market penetration, potential use cases for Phase II/III, and AFRL priorities that ANSYS would likely be the ideal partner. A letter of support was obtained as the Phase II proposal was developed, and the bulk of integration effort on the FEM side will focus on this platform. Initial discussions about available interfaces for extension sound promising, but it was not possible to obtain access before a Phase II project to do any initial investigation of integration.

Due to their proprietary nature, it remains a priority to develop some integration with an open source FEM solver, such as Albany. This would enable the team to offer an open source reference implementation, and something that can be used as a demonstration of capabilities in the framework free of any licensing restrictions. The interface to ANSYS would remain closed, and having some integration with two solvers would ensure that the abstraction of the FEM solver is effective.

5.0 CONCLUSIONS

The flow of data represents one of the major challenges in multiscale modeling, especially important if we are to develop user-friendly tools likely to be used in structural design. The links between length scales end up being quite specific to the design problem, and so within the confines of a Phase II project it is necessary to develop a framework around important and commercially relevant design problems. This is likely to offer a platform whose value will be seen, that would then attract further funding to extend to new systems using a software services model.

FEM solvers tend to have integration points to extend their internal models. The use of MPI or socket communication with the solvers would offer the most efficient integration, but would also be the most intensive. The development of a framework, with abstraction of the specific FEM solver and material code, will result in a flexible platform where a staged integration can take place. Abstraction would mitigate vendor lock-in, and offer agility to use the best tools for the design problem. Facilities such as materials databases, smart caching, communication with FEM solvers, coordination of materials simulations, and so forth, can be developed in the framework.

The development of a framework would also offer a place to store a "wider" view of the objects managed, so instead of attempting to store parameters in the FEM code, or materials code, the framework can track both length scales. This information can be handed off to traditional tools with the data they support, and enhanced interaction can take place in tools such as CMB. It looks likely that commercial design environments offer facilities that could open CMB or other graphical windows in order to display this enhanced information.

The main challenges in developing such a framework were identified in Phase I, and fed into the proposal for Phase II development. Initial integration with several major components has now been demonstrated, and these early prototypes indicate that they are amenable to being integrated in larger, automated workflows. It became clear when rerunning steps manually that there is tremendous value in automating these workflows, and modeling use cases as extended computational pipelines will make it easier to model these systems.

List of Acronyms, Abbreviations, and Symbols

AFRL	Air Force Research Laboratory
API	Advanced Programming Interface
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CFD	Computational Fluid Dynamics
CMB	Computational Model Builder
CMC	Ceramic Matrix Composites
COTS	Commercial Off The Shelf Software
DOD	Department of Defense
ERDC	US Army Engineer Research and Development Center
FEA	Finite Element Analysis
FEM	Finite Element Modeling
GE	General Electric
GTF	Geared Turbo Fan
GUI	Graphical User Interface
HPC	High Performance Computing
I/O	Input/Output
IP	Intellectual Property
LDRD	Laboratory Directed Research and Development
MTR	Micro-Textured Regions
R&D	Research and Development
SBIR	Small Business Innovation Research
SMTK	Simulation Modeling Toolkit
TEM	Transmission Electron Microscopy
VTK	Visualization Toolkit